

In the Claims:

1. (original) A method of tagging results of an XML query over a relational database, said method comprising:
generating a tagger tree graph from said XML query, each node of said tagger tree graph comprising a tagger operator, each tagger operator having a parse tree associated therewith;
calling each tagger operator in accordance with a structure of said tagger tree graph, and
evaluating said parse trees associated with each called tagger operator to tag results of said XML query over said relational database.

2. (original) A method of tagging results of an XML query over a relational database, as per claim 1, wherein said tagger node graph has a top-most tagger operator and a plurality of lower-most tagger operators, said calling and evaluating steps further comprising:

a. starting with said top-most tagger operator, each tagger operator implementing a method to request results from inputs to said tagger operator, said method causing lower-level tagger operators connected to said inputs to be called;

b. starting with said lower-most tagger operators, each called tagger operator returning intermediate tagged results to a higher-level connected tagger operator upon evaluating said associated parse tree;

performing steps a and b until an end of said results of said XML query is reached, and said top-most tagger operator producing tagged output XML of said results of said XML query.

3. (original) A method of tagging results of an XML query over a relational database, as per claim 1, wherein said tagger operators comprise any of a tagger input operator, a tagger scalar operator or a tagger aggregate operator.

4. (currently amended) A method of tagging results of an XML query over a relational database, as per claim 1, wherein said tagger tree graph includes a tagger ~~input~~ operator for each level in a result XML tree of said XML query.

5. (original) A method of tagging results of an XML query over a relational database, as per claim 4, wherein said tagger input operators execute in a sorted outer union mode.

6. (original) A method of tagging results of an XML query over a relational database, as per claim 5, wherein said tagger input operators comprise a shared tagger row stream.

7. (original) A method of tagging results of an XML query over a relational database, as per claim 4, wherein said tagger input operators execute in a node strip mode.

8. (original) A method of tagging results of an XML query over a relational database, as per claim 7, wherein each of said tagger operators comprises a tagger row stream.

9. (original) A method of tagging results of an XML query over a relational database, as per claim 1, wherein each tagger operator performs any of a cr8_elem, a cr8_attr, a cr8_attr_list, a cr8_fragments or a cr8_fragment_list function.

10. (original) A method of tagging results of an XML query over a relational database, as per claim 1, wherein each tagger operator implements a next method to produce a result row.

11. (original) A method of tagging results of an XML query over a relational database, as per claim 1, said method further comprising:

parsing said XML query;

transforming said XML queries into a language-neutral intermediate representation;

rewriting said language-neutral intermediate representation into an equivalent form easily

translated into an SQL query;

translating said equivalent form into one or more SQL queries over said relational database, and

executing said one or more SQL queries to produce said results of said XML query over said

relational database.

12. (currently amended) A method of tagging results of an XML query over a relational database, as per claim 11, wherein said tagger tree graph is generated from said equivalent form.

13. (currently amended) A method of tagging results of an XML query over a relational database, as per claim 11, wherein said tagger tree graph includes a tagger ~~input~~ operator for each node in a result XML tree of said XML query.

14. (original) A method of tagging results of an XML query over a relational database, as per claim 13, wherein said tagger input operators execute in a sorted outer union mode and said translating step produces a single SQL query to produce a single sorted outer union relational database result.

15. (original) A method of tagging results of an XML query over a relational database, as per claim 14, wherein said tagger input operators comprise a shared tagger row stream.

16. (original) A method of tagging results of an XML query over a relational database, as per claim 13, wherein said tagger input operators execute in a node strip mode and said translating step produces a set of SQL queries to produce a set of node strip relational database results.

17. (original) A method of tagging results of an XML query over a relational database, as per claim 16, wherein each of said tagger operators comprises a tagger row stream.

18. (original) A method of tagging results of an XML query over a relational database, as per claim 11, wherein said tagger operators comprise any of a tagger input operator, a tagger scalar operator or a tagger aggregate operator.

19. (original) A method of tagging results of an XML query over a relational database, as per claim 11, wherein a number of relational database tables of said relational database are mapped to a number of virtual XML documents and said XML queries are issued over said virtual XML documents.

20. (original) A method of tagging results of an XML query over a relational database, as per claim 1, wherein said method operates over a distributed computing network.

21. (original) A method of tagging results of an XML query over a relational database, as per claim 20, wherein said method operates over the Internet.

22. (original) A system for tagging results of an XML query over a relational database, said system comprising:

a tagger runtime;

a tagger tree graph generated from said XML query, each node of said tagger tree graph comprising a tagger operator;

a parse tree associated with each tagger operator, and

wherein said tagger runtime calls each tagger operator in accordance with a structure of said tagger tree graph and evaluates said parse trees associated with each called tagger operator to tag results of said XML query over said relational database.

23. (original) A system for tagging results of an XML query over a relational database, as per claim 22, wherein said tagger node graph has a top-most tagger operator and a plurality of lower-most tagger operators, and to perform said calling and evaluating, said tagger runtime further:

a. starting with said top-most tagger operator, causing each tagger operator to implement a method to request results from inputs to said tagger operator, said method causing lower-level tagger operators connected to said inputs to be called;

b. starting with said lower-most tagger operators, causing each called tagger operator to return intermediate tagged results to a higher-level connected tagger operator upon evaluating said associated parse tree;

performing steps a and b until an end of said results of said XML query is reached, and upon reaching an end of said results of said XML query, causing said top-most tagger operator to produce a tagged output XML document of said results of said XML query.

24. (original) A system for tagging results of an XML query over a relational database, as per claim 22, wherein said tagger operators comprise any of a tagger input operator, a tagger scalar operator or a tagger aggregate operator.

25. (original) A system for tagging results of an XML query over a relational database, as per claim 22, wherein said tagger graph includes a tagger input operator for each node in a result XML tree of said XML query.

26. (original) A system for tagging results of an XML query over a relational database, as per claim 25, wherein said tagger input operators execute in a sorted outer union mode.

B 27. (original) A system for tagging results of an XML query over a relational database, as per claim 26, wherein said tagger input operators comprise a shared tagger row stream.

28. (original) A system for tagging results of an XML query over a relational database, as per claim 25, wherein said tagger input operators execute in a node strip mode.

29. (original) A system for tagging results of an XML query over a relational database, as per claim 28, wherein each of said tagger operators comprises a tagger row stream.

30. (original) A system for tagging results of an XML query over a relational database, as per claim 22, wherein each tagger operator performs any of a cr8_elem, a cr8_attr, a cr8_attr_list, a cr8_fragments or a cr8_fragment_list function.

31. (original) A system for tagging results of an XML query over a relational database, as per claim 22, wherein each tagger operator implements a next method to produce a result row.

32. (original) A system for tagging results of an XML query over a relational database, as per claim 22, said system further comprising:

a parser, said parser parsing said XML query and transforming said XML queries into a language-neutral intermediate representation;

a rewrite engine, said rewrite engine rewriting said language-neutral intermediate representation into an equivalent form easily translated into an SQL query;

a translator, said translator translating said equivalent form into one or more SQL queries over said relational database, and

an RDBMS, said RDBMS executing said one or more SQL queries to produce said results of said XML query over said relational database.

33. (original) A system for tagging results of an XML query over a relational database, as per claim 32, wherein said tagger graph is generated from said equivalent form.

34. (original) A system for tagging results of an XML query over a relational database, as per claim 32, wherein said tagger graph includes a tagger input operator for each node in a result XML tree of said XML query.

35. (original) A system for tagging results of an XML query over a relational database, as per claim 34, wherein said tagger input operators execute in a sorted outer union mode and said

translator produces a single SQL query to produce a single sorted outer union relational database result.

36. (original) A system for tagging results of an XML query over a relational database, as per claim 35, wherein said tagger input operators comprise a shared tagger row stream.

37. (original) A system for tagging results of an XML query over a relational database, as per claim 34, wherein said tagger input operators execute in a node strip mode and said translator produces a set of SQL queries to produce a set of node strip relational database results.

38. (original) A system for tagging results of an XML query over a relational database, as per claim 37, wherein each of said tagger operators comprises a tagger row stream.

39. (original) A system for tagging results of an XML query over a relational database, as per claim 32, wherein said tagger operators comprise any of a tagger input operator, a tagger scalar operator or a tagger aggregate operator.

40. (original) A system for tagging results of an XML query over a relational database, as per claim 32, said system further comprising:

a schema mapper, said schema mapper mapping a number of relational database tables of said relational database to a number of virtual XML documents, and
an XML-QL engine, said XML-QL engine issuing said XML queries over said virtual XML documents.

41. (original) A system for tagging results of an XML query over a relational database, as per claim 22, wherein said system operates over a distributed computing network.

42. (original) A system for tagging results of an XML query over a relational database, as per claim 41, wherein said system operates over the Internet.

43. (original) A system for tagging results of an XML query over a relational database, as per claim 22, wherein said tagger runtime operates outside an RDBMS.

44. (original) A system for tagging results of an XML query over a relational database, said system comprising:

means for generating a tagger tree graph from said XML query, each node of said tagger tree graph comprising a tagger operator, each tagger operator having a parse tree associated therewith;
means for calling each tagger operator in accordance with a structure of said tagger tree graph,
and
means for evaluating said parse trees associated with each called tagger operator to tag results of said XML query over said relational database.

45. (original) A computer program product comprising a machine-readable medium including computer readable program code therein for tagging results of an XML query over a relational database comprising:
computer readable program code generating a tagger tree graph from said XML query, each node of said tagger tree graph comprising a tagger operator, each tagger operator having a parse tree associated therewith;

computer readable program code calling each tagger operator in accordance with a structure of said tagger tree graph, and
computer readable program code evaluating said parse trees associated with each called tagger operator to tag results of said XML query over said relational database.

46. (original) A computer program product comprising a machine-readable medium including computer readable program code therein for tagging results of an XML query over a relational database as per claim 45, wherein said generated tagger node graph has a top-most tagger operator and a plurality of lower-most tagger operators, said calling and evaluating computer readable program code further comprising:

computer readable program code for performing:

a. starting with said top-most tagger operator, each tagger operator requesting results from inputs to said tagger operator, said request causing lower-level tagger operators connected to said inputs to be called;

b. starting with said lower-most tagger operators, each called tagger operator returning intermediate tagged results to a higher-level connected tagger operator upon evaluating said associated parse tree;

performing steps a and b until an end of said results of said XML query is reached, and

said top-most tagger operator producing tagged output XML of said results of said XML query.
